Balancing Value and Risk In Information Sharing Through Obfuscation

Supriyo Chakraborty, Kasturi Rangan Raghavan, Mani B. Srivastava
University of California, Los Angeles

Chatschik Bisdikian
IBM T.J. Watson Research Center

Lance M. Kaplan
US Army Research Laboratory

***Abstract*—Fast-paced data-to-decision systems are heavily dependent on the reliable sharing of sensor-derived information. At the same time a diverse collection of sensory information providers would want to exercise control over the information shared based on their perception of the risk of possible misuse due to sharing and also depending on the consumer requirements. To attain this utility vs. risk trade-off, information is subjected to varying but deliberate quality modifying transformations which we term as obfuscation. In this paper, treating privacy as the primary motivation for information control, we highlight initial considerations of using feature sharing as an obfuscation mechanism to control the inferences possible from shared sensory data. We provide results from an activity tracking scenario to illustrate the use of feature selection in identifying the various trade-off points.**

***Index Terms*—Privacy, Value of Information, Utility, Feature Sharing, Mutual Information**

I. Introduction

We observe our physical world to gain an understanding of it, to draw informed decisions and drive effective actions. For example, we may observe vehicles to infer driving patterns and to improve commute times. We may collect intelligence (from both hard and soft sensors) about people, asset concentration and movement patterns to draw inferences about eminent hostile actions and prepare against. We may collect vital signs to monitor human activities and conditions to provide remote healthcare, or to conclude about the effectiveness of a drug therapy. In addition to the reasons "declared" (maybe as part of a sharing contract) for collecting observations, these could be used to identify and locate a person and learn about his lifestyle, or localize the sensory resources and their capabilities owned by a military coalition partner unwilling to share this information.

All the above are examples of data-to-decision systems and applications that exploit the collective power of an ever increasing corpus of low-cost sensors deployed by "us" or "others." These systems and applications collect and intelligently distribute the sensory information, or the outcome and its processing, to the proper edge elements that are in need of the aforementioned understanding of the world to make decisions and take actions.

Sensory information possesses innate properties (e.g., accuracy, latency, provenance), summarily referred to as *quality of information* (QoI), that result from the sensing and communication operations and, also, carries value (VoI *value of information*) that depends on its use and user context [1]. In addition, information may carry value for its provider as well, such as sharing it for monetary benefit, for mitigating congestion, for aiding a military coalition partner execute its task better, for contributing to medical research for curing a specific disease and so on.

Therefore, it is natural to expect that a provider may seek to exercise control over the information shared. This could be aimed towards protecting aspects of his information producing infrastructure from being revealed, such as its sensing capabilities, sensor locations, etc., that differentiate him from other providers; or simply to protect certain inferences from being revealed to the information consumer. Such revealing could be possible by correlating the shared information with additional information available to the consumer, such as correlating tracking information about an object received from the provider with independently obtained information about the object's track to infer about the capabilities of the provider. Due to the concerns associated with the value that the shared information has on its provider, we will refer to this value as the *risk of information* (RoI).

The QoI, VoI and RoI are aspects associated to an information product that can be traded-off to achieve a balance between the utility gained by the information consumers and risk experienced by the information providers. We will refer to as *obfuscation* any information processing done deliberately by the provider with the objective to attain this balance. QoI in this case plays the role of a catalyst whose increase or decrease affects consumer utility and provider risk. Fig. 1 illustrates this point showing a piece of information of a given quality level (QoI) and corresponding value (VoI) to a consumer. However, this information carries risks to the provider due to unintended uses of it by the consumer. The provider will make judicious use of obfuscation to continue providing value to the consumer while containing its own risk.

Traditionally associated to privacy, in this paper, we present an early investigation of a broader view of obfuscation as a means to manage inferences from sensory information that providers share with consumers. With a piece of information having many uses, hence, being of varying value in different use contexts, we take an approach of applying obfuscation at appropriate representations of information, e.g., obfuscating a

This work has been accepted for publication at the 15^{th} International Conference on Information Fusion (FUSION 2012).

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
20 SEP 2012		**00-00-2012 to 00-00-2012**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Balancing Value and Risk In Information Sharing Through Obfuscation	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **IBM T.J. Watson Research Center,19 Skyline Drive,Hawthorne,NY,10532**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
Presented at the Annual Conference of International Technology Alliance (ACITA) 2012 held 17-21 Sep in Southampton, UK. U.S. Government or Federal Rights License

14. ABSTRACT
Fast-paced data-to-decision systems are heavily dependent on the reliable sharing of sensor-derived information. At the same time a diverse collection of sensory information providers would want to exercise control over the information shared based on their perception of the risk of possible misuse due to sharing and also depending on the consumer requirements. To attain this utility vs. risk trade-off, information is subjected to varying but deliberate quality modifying transformations which we term as obfuscation. In this paper, treating privacy as the primary motivation for information control, we highlight initial considerations of using feature sharing as an obfuscation mechanism to control the inferences possible from shared sensory data. We provide results from an activity tracking scenario to illustrate the use of feature selection in identifying the various trade-off points.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**8**	

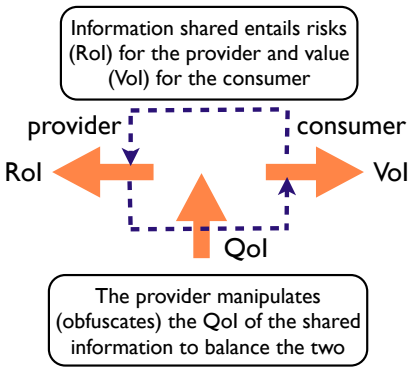

Fig. 1. Using QoI to balance the RoI of the provider and VoI of the consumer.

select set of features of a signal. The selection of the information representation and obfuscation seeks to steer inferences that consumers can make towards a white-list of inferences that carry low risk to the provider and away from a black-list of high-risk inferences, to simultaneously provide utility to the consumer while lowering provider risk. The paper builds on our the earlier short paper[1] [2] where we highlighted for the first time the interplay between QoI, RoI, VoI and the role that obfuscation has in balancing between QoI and RoI via QoI. We also introduced the activity tracker example case as a mean to demonstrate the use of black and white list of inferences for driving obfuscation using features of sensory signal. In this paper, we further expand upon these concepts and, in addition, we introduce a number of design alternatives for building an obfuscation framework residing at different points on the path between an information provider and consumer. We use these alternatives to argue the benefits in building the framework based on manipulating features of sensory signals. Finally, we introduce a number of system level architectures for building the obfuscation framework. We highlight the pros and cons in building the framework for a networked system of mobile users by placing the obfuscation functionality at various places from a user device to the cloud, or in between.

The rest of the paper is organized as follows. In Section II, we motivate the use of inferences as the "currency" for expressing the risk vs. value tradeoff problem. The possible obfuscation choice and operating points are discussed in Section III. An obfuscation framework implementing the obfuscation strategy is discussed in Section IV. Depending on the different data flow scenarios between providers and consumers (or applications requesting data), the placement of the obfuscation framework could vary. These choices are analysed in Section V. Related work is summarized in Section VI. The feature sharing approach similar to one presented in [2] is shown in Section VII and we demonstrate its efficacy using an example activity tracking scenario in Section VIII. A discussion on the approach and future directions is presented in Section IX.

[1]http://nesl.ee.ucla.edu/document/show/408

II. Features and Inferences

In this section, we use information privacy as the primary motivation for information control and outline various elements of a feature sharing strategy as an obfuscation mechanism.

A. Using Inferences as Primitives

We define an inference as an estimate of the current behavioral state of a provider. For example, a person can be either smoking or not smoking. An inference function would use sensory data to estimate the current state of the provider. If the provider wants to keep his smoking habit private then the scheme should transform the data to prevent the consumer from accurately estimating the smoking states of the provider.

From a consumer's perspective, the shared data is useful for drawing various inferences. From a provider's perspective, on the other hand, some of these inferences are acceptable, mutually agreed upon and contribute to consumer's utility whereas others are more private and contribute to overall risk [3]. Traditionally, obfuscation mechanisms are focussed on reducing a specific form of provider risk, one which arises from disclosure of private information. The implementation of these mechanisms include addition of structured noise to the data for reducing the QoI. The RoI is derived in terms of the reconstruction error of the original signal from the obfuscated data [4], [5]. However, these mechanisms overlook the fact that many of the private inferences can be drawn just as accurately from partially reconstructed signals or ones with low signal-to-noise ratio [6]. In other words, we posit that it is the *inference quality* and not the signal quality that should be the privacy metric to focus on.

B. Problem Definition

With the above insight, we consider the use of inference functions as the natural basis for expressing the utility and privacy requirements of a provider. We postulate the existence of a universe of computable inferences. A provider shares data such that the consumer is able to compute sufficiently well a subset of inferences from this universal set – maintaining utility. At the same time, the provider wants to protect a disjoint subset of inferences from being computed using the same data – reducing risk. We call the former subset the *white list* and the latter one the *black list*. Consideration for possible "gray lists" is left for future time. Armed with the white and black lists of inference functions, which are expected to be personalized, recipient-specific and context-dependent, the goal is to systematically design an obfuscation mechanism and evaluate whether sharing some data can be potentially divulging of private inferences incurring risk while simultaneously meeting the utility objective.

III. Possible Choices of Obfuscation

Information entering a data flow pipeline from providers to consumers is summarized into various forms at different stages of the pipeline. Initially, it exists as raw samples collected from sensors. Subsequently, features are extracted from the raw

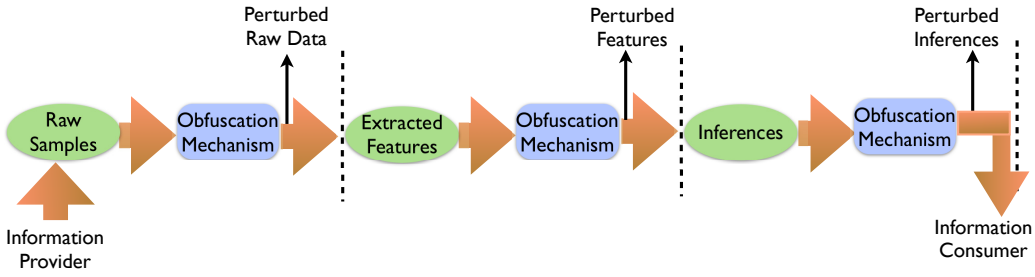

Fig. 2. Possible operating points of an obfuscation mechanism within a data flow pipeline from providers to consumers.

samples. Finally, the features are used by various algorithms to draw inferences which are used by the consumers. In such a pipeline as shown in Fig. 2, a QoI reducing obfuscation mechanism can operate at different points. Below we outline the possible options.

Access Control: Access control can occur at any of the three points in the pipeline. The provider is presented with a binary option of either allowing or denying access to the consumer on the requested information. Depending on the operating point, when granted permission, the consumer can access either raw data, features or inferences at the highest resolution and could maliciously use it for drawing unintended inferences. Thus, using access control the provider has two options – either to fully trust the consumer, or not trust at all. For example, current privacy practices for mobile phones mandate that an application during its installation declare the resources (e.g. raw sensory data) it needs access to. For instance, the manifest file on an Android based mobile system is one such enforcing mechanism. The privacy framework implements a policing mechanism to ensure that undeclared resources are never accessed. However, the consumer has complete access on resources for which permission has been granted. While fine-grained access control has been proposed in [7], [8], for a given access policy either utility or privacy is fully compromised.

Perturb Raw Data: The next logical step is to apply data transformations such as perturbation, suppression, generalization [9] on the raw data to change its "resolution" before sharing. For example, one could add noise drawn from a Gaussian distribution while preserving the statistical properties of the data. Other alternatives include adding structured noise [10] to selectively effect data features to simultaneously maintain both utility and privacy goals. The above strategy allows us to overcome the drawbacks of using a binary strategy as data could now be shared at different resolutions providing better control. The risk guarantee of a perturbation scheme is provided in terms of the reconstruction possibility of the original signal [11], [4], [10] from the perturbed data. However, a weakness of this strategy is that many private inferences could be made just as accurately with a lower signal-to-noise ratio – a problem not addressed by the reconstruction metric of privacy.

Feature Selection and Perturbation: Without an explicit notion of what constitutes a private inference it is difficult to devise perturbation techniques to protect against them. We posit that using inferences as the basic primitive applications and users should provide their white and black lists of utility and privacy preferences. An inference is typically computed using features such as mean, variance, FFT coefficients which are extracted from raw data. A natural alternative that emerges from the above analysis is to only share the features such that only the mutual information [12] with the white listed inferences is preserved and not the raw data – i.e. perform *feature selection* and *feature suppression* before sharing. While raw data could suffer from the curse of higher dimensionality and susceptibility to inference attacks [13] the process of feature extraction is dimensionality reducing [14], [15] and extracted features allows efficient computation of inferences (e.g. clustering, linear classifiers). In addition, sharing features allows better control over the amount of information that is being shared and lower the risk.

However, the disjoint white and black list inferences could have overlapping features. Therefore, we augment feature sharing with an additional step of *feature perturbation*, where we perturb the common features such that the black list inference quality is significantly degraded while maintaining the white list accuracy. However, there exists two caveats to this approach. First, it assumes that there exists prior knowledge regarding the mapping between inferences and the features they use. This can be justified through the observation that there are typical features that are used for drawing inferences from sensors commonly found on mobile devices. To illustrate, inferences from accelerometer data typically uses first-order statistics such as mean, variance or statistics over FFT coefficients. Second, applications need to be modified to work with features instead of raw data. This could be the cost of achieving better privacy as compared to sharing raw data.

Sharing Inferences: At the other extreme, the provider could locally compute all the inferences and share only a subset of obfuscated inferences. The problem with this approach is that the provider can at times be constrained in terms of both energy and computational resources. In addition, the algorithms used by the consumer for the inferences could be proprietary, therefore sharing them with the provider can be difficult.

IV. Obfuscation Framework

We will use a somewhat elementary data-to-decision exemplar to communicate the main thesis of our research that of a system-level obfuscation framework. Specifically, imagine an *activity tracker* smartphone app that uses the accelerometer to track the activity level of the user during a day. From a decision side, this information could be used, for example, to adjust the environmental (temperature, humidity, ambient light, music level,etc.) in the persons living quarters. Privacy and security measures available today mandate that applications declare what sensors they require during installation (or use) and a system framework provides a policing mechanism to ensure that undeclared resources are not accessed. However, suppose that the app developer maliciously included algorithms that also infer the type of a user's activity (sitting, sleeping, walking, running [16]) from the same accelerometer data. Since the system provides the app with raw accelerometer data, both system and user are oblivious to the true behavior of this application. In today's mobile platforms, users must either fully trust the application or choose to not use it at all. Unfortunately, as users become increasingly aware of the privacy risks associated with seemingly innocent apps, their negative decision stifles the adoption of the legitimate apps [3].

We propose that the system framework be augmented with an obfuscation framework. This is the core privacy-enabling layer that shapes sensor data and features extracted from sensor data before releasing it to applications that request it. The obfuscation framework is configured with a user-specified black list of inferences and only allows applications to access data to compute a set of declared and user-approved white list inferences. So, for example, the activity tracker may request accelerometer data from the system, but must specify that it intends to use it for determining user activity level to compute the total activity level. The obfuscation framework then allows features of the data that correspond to activity level (e.g. signal variance) to pass through, but blocks features relevant for inferring activity type (e.g. energy in specific frequency bins). We defer a more detailed handling of this application to Section VIII.

It must be emphasized that the obfuscation framework cannot guarantee absolute privacy, that is, against the unknown. Like its network counterpart, the obfuscation framework can only be as effective as its configuration. If an inference is not specified in the black list, or an inference function is incorrectly specified with respect to its features or its sensitivity to features, the obfuscation framework may fail. We introduce the notion of an obfuscation framework to express user privacy concerns in a coherent and general manner and provide a systematic analysis of the requirements of such a module within the sensing stack.

V. Interaction Scenarios

We start by describing our view of a mobile system used to collect and share sensory data. In Fig. 3(a), we show the system architecture of the Android mobile platform [17]. We combine the functionalities of the layers containing the Linux kernel, system libraries, Android runtime and the application framework, abstract it into a single block and call it the mobile platform. Applications running on the phone form part of the topmost layer. This view is shown in Fig. 3(b). Thus, data from various sensors are made available to the applications which are also the data consumers by the mobile platform in the middle. We assume that all the software components forming the mobile platform are trusted whereas the application layer (with third-party apps) is not trusted.

Using the above system view, Fig. 4 parts (a), (b) and (c) illustrate the interactions and the possible ways in which data flow can occur between the provider and the consumers. The mobile platform in the figure is demarcated by the dotted box.

- *Local:* Fig. 4(a) shows an untrusted application (app1) which runs locally on the mobile platform itself. It requires access to one or more sensors for providing its service. Since, the application is not trusted, the user would want to ensure that it is not able to make any of the blacklisted inferences using the provided data.
- *Direct-Cloud-Hosted:* Fig. 4(b) shows an application client which runs on the mobile system, but relies on a cloud-hosted server for making the inferences. The client collects data streams from various sensors and uploads them to the server. Inferences made at the server are then sent back to the client and the user.
- *Broker-Cloud-Hosted:* In Fig. 4(c), the interaction between the mobile system and the applications occurs through a trusted broker. A trusted broker is a server which is either owned by the user, or by a trusted third-party. User uploads data to the broker which in turn provides it to the cloud-hosted application for obtaining the desired service. Services such as [7], [18], [19] follow the trusted broker model.

Depending on the type of interaction the placement of the obfuscation framework could vary. In part (a), the framework needs to be placed between the mobile platform and the applications. Data streams passing through the framework layer are adequately transformed before they are handed over to the untrusted application. In part (b), the framework can be included into the client implementation in the application layer. While this would eliminate the need to make system level changes as required for part (a), it would involve modification of the code base for different clients. However, some of the changes can be application specific resulting in significant duplication of effort across multiple clients. In addition, the modified client code should be trusted and not leak information. In part (c) we could push the obfuscation framework to the trusted broker. This would eliminate the need for changes at both the application client as well as the system level. However, as mentioned above, these would be point solutions needing replication for all such brokers. Therefore, an additional layer as part of the trusted mobile architecture itself is the most effective way of implementing the obfuscation framework.

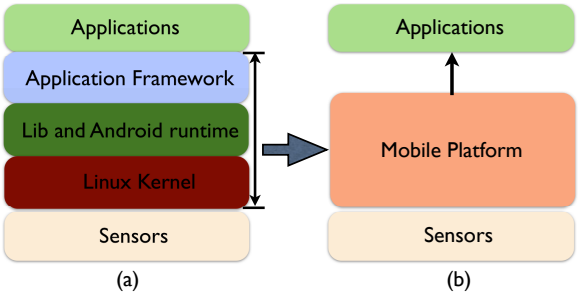

Fig. 3. Our View of a Mobile System.

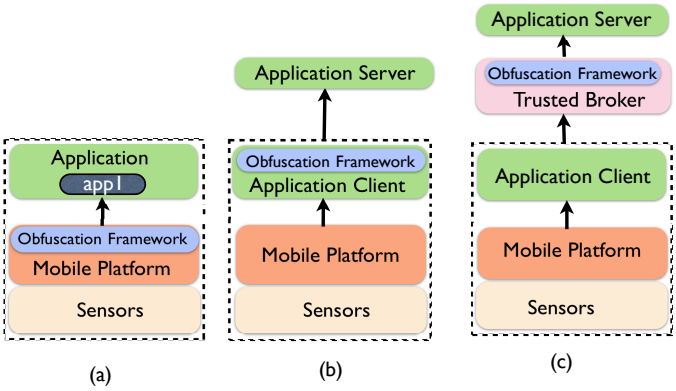

Fig. 4. Different Data Flow Paths from provider to consumer. Data shared with (a) locally running app (b) directly with cloud-hosted app (c) via trusted broker with cloud-hosted app.

VI. Related Work

While privacy challenges of sharing sensory data have been relatively less explored, there is a large body of work which addresses privacy concerns during disclosure of relational data from databases. We broadly categorize them into four categories. First, there is work that explores system architecture mechanisms for providing the source greater control over data dissemination, and includes systems such as PDV [7] and Locaccino [18] for sensor data, and others such as Virtual Individual Servers [20], Lockr [19], Persona [21] and Microsoft Health Vault [22] for more generic data records. Most of these provide relatively coarse-grained access control and limited conditioning of access upon context such as users location, activity, and behavior. However, the binary nature of access control mechanisms prevents one from achieving the utility aspect of the sharing problem. Second, there is work in the statistical database community that seeks to provide anonymity for an individual user by combining data from multiple users to achieve privacy measured by metrics such as k-anonymity [23], l-diversity [24], and t-closeness [25] using mechanisms such as perturbation, suppression, generalization etc. [9]. Recently, differential privacy [26] has emerged as a principled way of perturbing responses to statistical queries issued on databases. Protecting user identity is the main objective of these approaches, however, in most of our example scenarios we assume that the user identity is part of the shared data. In addition, all these approaches adopt a "privacy-first" policy and provide no explicit mechanism for expressing or achieving a desired utility objective. Third category comprises of recent advances in cryptography, particularly homomorphic encryption [27], which while not yet efficient in its full variant, can still be used in limited forms for processing sensor data by cloud-based intermediary services that are not fully trusted. Finally, and most related to our work are approaches that transform sensor data before sharing so as to provide privacy while preserving its utility, such as in [4], [11]. While offering point examples, these works do not provide a general approach and system architecture as we do. Moreover, the formulation of privacy and utility in current works is typically in terms of recipient's inability [4] and ability [11] respectively to reconstruct the signal, which is a very limiting concept compared to our inference-based approach.

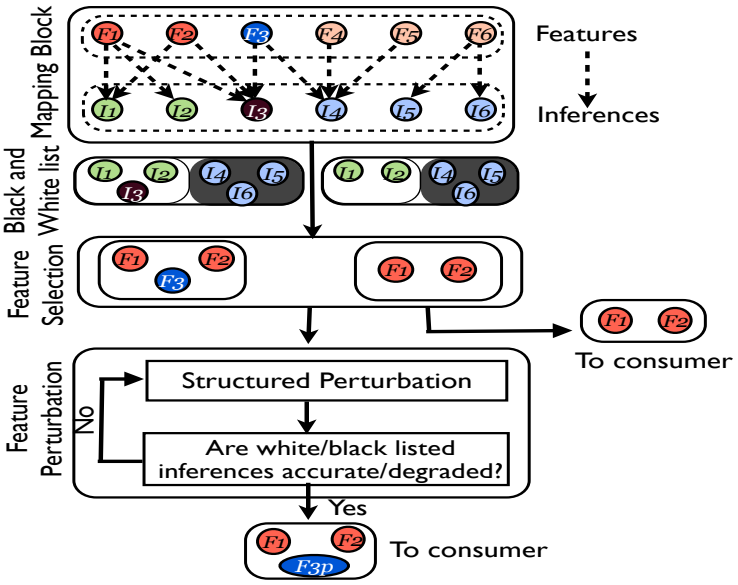

Fig. 5. A system realization of the feature sharing based obfuscation mechanism.

VII. Realization of an Obfuscation Mechanism

Determining the optimal set of features to share forms the crux of the solution approach. Intuitively, the selected subset of features should allow accurate computation of white listed inferences and contain no information for computing the black listed inferences. This motivates the use of mutual information between the features and inferences as a selection metric. The optimization problem therefore is to select a subset of features such that the mutual information with the white listed inferences is maximized and that with the black listed inferences is simultaneously minimized. Ideally, if the joint distribution between a finite set of features and inferences is known, we can use it to evaluate the mutual information between all possible combinations of features and inferences and select one which satisfies the optimization criterion. However, there are two problems with this ideal scenario. First, it is hard to compute the joint distribution because of inadequate training data. Second, evaluating over

all possible feature combinations could be of exponential complexity. To address the first issue, we are guided by Fano's inequality [12] and use misclassification as an approximation to the mutual information. However, the limitation of this is that the privacy guarantee is for the state-of-the-art inference algorithm used for classification. We are currently working on a suitable heuristic for selecting the right subsets of features to evaluate misclassification against. Finally, after evaluation if the subset of features selected is completely disjoint from the features required by the black listed inference we call it *feature selection*. If there exists an overlap, we pass it through a *feature perturbation* block to further increase the black list misclassification error while maintaining the misclassification for the white list before sharing.

Figure 5 shows two applications that use different features (top row of the mapping block) for their respective white and black list configurations. Both applications black list inferences I_4, I_5 and I_6 (bottom row of the mapping block) and could therefore strip the data of features F_4, F_5 and F_6. For the application on the right, choosing to share features F_1 and F_2 is adequate to compute the white list inferences I_1 and I_2. However, for the application on the left, feature F_3 is used in the black listed inference I_4 and the white listed inference I_3. In order to meet utility and privacy requirements the system shares a perturbed version of F_3: F_{3p}. Since I_3 also uses features F_1 and F_2, which are shared unmodified, the utility of inference I_3 may not degrade substantially because of the perturbed feature F_{3p}. Whereas, since the inference I_4 needs F_3, F_4, and F_5, the latter two of which were removed from the data, the black listed inference I_4 only has F_{3p} to its disposal. The degree to which an inference is degraded because of feature perturbation depends on the sensitivity of the inference to the feature and the form of perturbation being applied.

VIII. Activity Tracking Application

It is well known that activities can be inferred from accelerometer data [16], [28]. But, there are several privacy concerns that could arise. First, depending on current context (e.g., location) a user might not be comfortable sharing his activity with a third party application [3]. Second, from a record of activity across long time scales, one can model sensitive behavioral patterns [29]. Consider a simple application which tracks the intervals of time during which a person is active. One can imagine that this application polls the accelerometer to infer the level of activity (either active or inactive) over a short period of time. The idea here is to permit activity detection (the white list) and prohibit activity classification (the black list). Activity classification refers to a set of specific activities (e.g., walking, running, cycling) a user would like to restrict, so as to prevent behavioral mining. We use this application to illustrate the described feature sharing approach. Inferences from accelerometer data typically work on a set of simple features extracted from the waveform. If we can establish the set of features that can be used to perform activity classification, then the feature selection technique described in Section II can be used to preserve privacy. There are various ways to establish this mapping between features and inferences. One possible approach is to use machine learning to train an activity classifier based on a set of features. If we do this exhaustively for every combination of features this empirically establishes which sets of features can classify activity with high accuracy.

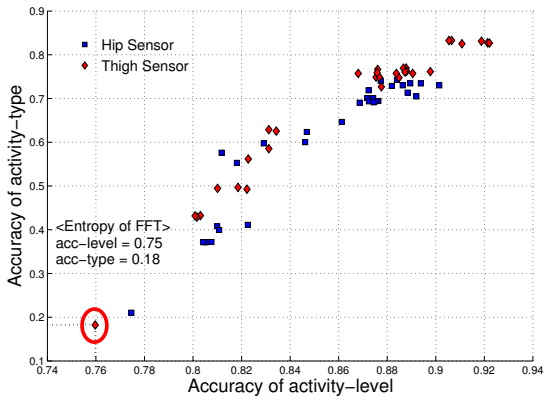

Fig. 6. Characterizing utility vs. privacy in terms of classification accuracies for various combination of features.

To test our approach, we have used an annotated acceleration dataset collected under controlled condition from 20 individuals [28]. Accelerometers were positioned on the the hips and thighs of the individuals and 20 everyday activities were considered as possible inferences. These activities include regular tasks like walking and running, but also relatively inactive tasks such as relaxing and standing still. This way, each of the 20 activities can be mapped to active or inactive, giving us annotations for both the white listed activity detection as well as the black listed activity recognition. We implemented five features extracted from accelerometer readings: *mean*, *variance*, *correlation between the axes* (considering only two axes), *entropy of FFT coefficients* and *energy of FFT coefficients*. For each set of features we learn a cross-validated C4.5 decision tree and use it to evaluate the accuracy of activity classification for that set [16]. We can use the same method to evaluate the accuracy for activity detection. This allows us to characterize the privacy vs. utility trade-off in terms of classification accuracies.

We interpret the complement of the activity recognition accuracy as privacy. Similarly, the activity detection accuracy is interpreted as utility. Thus, we can characterize a particular choice of feature combination as a privacy vs. utility trade-off point. Fig. 6 visualizes the various trade-off choices. For example, the circled point at the lower left corner means that 75% of the time the *entropy of FFT coefficients* feature can be used to correctly identify the activity level and only 18% of time identify the actual activity. Clearly, given the results in the figure, it would be desirable to operate at this point, which corresponds to high enough accuracy in the activity level and low accuracy in detecting activity type.

IX. Discussion and Future Work

Data-to-decision solutions will involve a multitude of sensory information providers, processors and consumers. Balancing value gained by sharing desired information with the risk from unintended uses of (or, inferences made with) this information will be a significant contributor to the effective deployment and operation of multiparty data-to-decision solutions.

In this paper, we have argued on the role that we expect obfuscation to play in enabling this balance. Specifically, considering the risk in privacy in the context of a sensor-enabled remote activity tracking application as an exemplar use case, we have argued that inference functions are the common ground through which providers can coherently communicate with the sensory information sources in mobile devices to express privacy concerns. In addition, we propose that mobile platforms should incorporate finer-grained access to sensor data by exploiting knowledge of common features that could be extracted from the data. This enables better presentation of intent, in the sense that when applications or consumers specify the set of useful inferences, they indirectly point to a set of features that need to be shared for those inferences to be made successfully.

In the simplest case, where an application requests features that do not reveal much about blacklisted inferences, the obfuscation framework can directly share those features. At the other extreme, when the requested features have some potential for drawing the blacklisted inferences a strategy of feature suppression, where only select features of the full requested set are shared, works. More realistically, however, are scenarios where the features from the white and black listed inferences overlap. The topic of this section is a more systematic, though as yet not fully developed approach that uses random projections as a proxy for features to resolve this issue.

A. Random Projections; Privacy-First

The approaches to inference privacy described insofar have two practical drawbacks. First, it is assumed that the obfuscation framework has complete knowledge of the mapping between inferences and features. It is through this mapping configuration that the obfuscation framework can ensure both privacy and utility. Second, the obfuscation framework cannot guard against privacy attacks from an arbitrary classifier, that is, for which the inference function is unknown. It is arguable that a malicious entity may be able to learn a different and undisclosed model and may use features not yet configured in the obfuscation framework.

Viewing inference privacy from a privacy-first angle, we propose sharing projections of the features instead of the features themselves [30]. That is, we transform the features to another space before sharing. Now, to ensure that privacy is maintained the transformation is kept private and is known only to the data provider. In order to meet the utility goals, the data provider only furnishes training labels so that the data consumer can learn a model of the inference from these projected features and labels rather than use a pre-built model function on the features directly. For this to work, however, the key property of the transformation that needs to be established is that of preserving pair-wise distances between points in the original feature space and the projected feature space.

Fortunately, when the transformation is derived from randomly generated basis vectors drawn from an i.i.d. normal distribution, the Johnson Lindenstrauss lemma states this property holds with high probability when the dimensionality of the new projected feature space satisfies a size constraint [31]. Thus, what was inferable via linear methods in the original feature space will still be inferable from these random projections. Furthermore, any inferences that can be drawn are limited to those for which the user provides ground truth training labels. In effect, the burden is placed on the application to learn a model from these random projections with the user-provided ground truth as labels.

This approach eliminates the drawbacks of the previous ones, but introduces some practical considerations of its own. The primary concern is that the application now must learn a model, local to the device. This may place a significant burden on the application. On the other hand, we can guarantee privacy when there is no side information, because the only inference that can be learned from the randomly projected features are ones explicitly trained for. It is worth noting that since the random transformation is never shared with any party, the dimensionality of the projected space can be arbitrarily large without fear of leakage from inversion attacks.

We hope a more thorough exploration of this type of approach will lead to a better theoretical foundation for providing inference privacy.

Acknowledgement

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defense under Agreement Number W911NF-06-3-0001 and by the NSF under award #0910706. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defense or the U.K. Government or the NSF. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

[1] C. Bisdikian, L. Kaplan, M. Srivastava, D. Thornley, D. Verma, and R. Young, "Building principles for a quality of information specification for sensor information," in *12th Int'l Conf. on Information Fusion (FUSION'09)*, 2009.

[2] S. Chakraborty, K. R. Raghavan, M. Srivastava, C. Bisdikian, and L. Kaplan, "An obfuscation framework for controlling value of information during sharing," ser. IEEE Statistical Signal Processing Workshop (Submitted), 2012.

[3] A. Raij, A. Ghosh, S. Kumar, and M. Srivastava, "Privacy risks emerging from the adoption of innocuous wearable sensors in the mobile environment," ser. CHI, 2011.

[4] H. Ahmadi, N. Pham, R. Ganti, T. Abdelzaher, S. Nath, and J. Han, "Privacy-aware regression modeling of participatory sensing data," ser. SenSys, 2010.

[5] L. Sankar, S. R. Rajagopalan, and V. Poor, "A theory of utility and privacy of data sources," ser. ISIT, 2010.

[6] S. Chakraborty, H. Choi, and M. Srivastava, "Demystifying privacy in sensory data: A QoI based approach," ser. IQ2S, 2011.

[7] M. Mun, S. Hao, N. Mishra, K. Shilton, J. Burke, D. Estrin, M. Hansen, and R. Govindan, "Personal data vaults: A locus of control for personal data streams," ser. ACM CoNEXT, 2010.

[8] H. Choi, S. Chakraborty, Z. M. Charbiwala, and M. B. Srivastava, "Sensorsafe: a framework for privacy-preserving management of personal sensory information," ser. SDM, 2011.

[9] B. Fung, K. Wang, R. Chen, and P. Yu, "Privacy-preserving data publishing: A survey on recent developments," *ACM Computing Surveys*, vol. 43, no. 4, 2010.

[10] Y. Kim, E. Ngai, and M. Srivastava, "Cooperative state estimation for preserving privacy of user behaviors in smart grid," *SmartGridComm*, 2011.

[11] S. R. Rajagopalan, L. Sankar, S. Mohajer, and H. V. Poor, "Smart meter privacy: A utility-privacy framework," *CoRR*, vol. abs/1108.2234, 2011.

[12] T. Cover and J. Thomas, *Elements of information theory*. New York, NY, USA: Wiley-Interscience, 1991.

[13] C. C. Aggarwal, "On k-anonymity and the curse of dimensionality," ser. VLDB '05, 2005, pp. 901–909.

[14] "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1-3, pp. 37 – 52, 1987.

[15] Pierre and Comon, "Independent component analysis, a new concept?" *Signal Processing*, vol. 36, no. 3, pp. 287 – 314, 1994.

[16] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava, "Using mobile phones to determine transportation modes," *ACM Trans. Sen. Netw.*, March 2010.

[17] http://http://developer.android.com/guide/basics/what-is-android.html.

[18] E. Toch, J. Cranshaw, P. Hankes-Drielsma, J. Springfield, P. G. Kelley, L. Cranor, J. Hong, and N. Sadeh, "Locaccino: a privacy-centric location sharing application," in *Proc. of 12th ACM International Conference on Ubiquitous Computing*, 2010.

[19] A. Tootoonchian, S. Saroiu, Y. Ganjali, and A. Wolman, "Lockr: better privacy for social networks," in *Proc. of the 5th international conference on Emerging networking experiments and technologies*, 2009.

[20] R. Cáceres, L. Cox, H. Lim, A. Shakimov, and A. Varshavsky, "Virtual individual servers as privacy-preserving proxies for mobile devices," in *In Proc. of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds*, 2009.

[21] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin, "Persona: an online social network with user-defined privacy," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4, 2009.

[22] http://www.healthvault.com.

[23] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 2002.

[24] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Trans. Knowl. Discov. Data, 2007*, March 2007.

[25] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity." in *ICDE*, 2007.

[26] C. Dwork, "Differential privacy," ser. ICALP, 2006.

[27] C. Gentry, "Fully homomorphic encryption using ideal lattices," ser. STOC, 2009.

[28] L. Bao and S. Intille, "Activity recognition from user-annotated acceleration data," *Pervasive Computing*, vol. LNCS 3001, 2004.

[29] "Reality mining," http://reality.media.mit.edu.

[30] K. Liu, H. Kargupta, and J. Ryan, "Random projection-based multiplicative data perturbation for privacy preserving distributed data mining," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 1, pp. 92–106, 2006.

[31] S. Dasgupta and A. Gupta, "An elementary proof of a theorem of johnson and lindenstrauss," *Random Struct. Algorithms*, vol. 22, pp. 60–65, January 2003.